The Pennsylvania State University

The Graduate School

Department of Mechanical Engineering

TURBINE ENGINE DIAGNOSTICS (TED) FOR ARMY TANK APPLICATIONS

A Thesis in

Mechanical Engineering

by

Mark J. Pincoski

Submitted in Partial Fulfillment

of the Requirements

for the Degree of

Master's of Science

December 1996

We approve the thesis of Mark J. Pincoski.

Date of Signature

Vigor Yang
Professor of Mechanical Engineering
Thesis Advisor

11-12-96

Asok Ray
Professor of Mechanical Engineering

11-13-96

Richard C. Benson
Professor of Mechanical Engineering
Head of the Department of Mechanical Engineering

11-13-96

Abstract
Turbine Engine Diagnostics (TED) for Army Tank Applications
Mark J. Pincoski
M.S. December 1996
The Pennsylvania State University
Vigor Yang, Thesis Advisor

Turbine engine diagnostics have been vastly improved with the use of Artificial Intelligence (AI) techniques such as expert systems, artificial neural networks and fuzzy logic. A typical system that is using artificial intelligence to improve its diagnostic capabilities is the Army's Turbine Engine Diagnostic (TED) program for the M1 Abram's AGT-1500 turbine engine. TED is a diagnostic expert system that assists the M1 Abrams mechanic. The system provides assistance during engine inspection and troubleshooting. It provides detailed information about the most frequently used maintenance procedures. It has an automated parts ordering system. Finally it has a diagnostics tool capable of monitoring the engine's electronic signals.

Table of Contents

List of Figures

List of Tables

Chapter 1

Introduction

As turbine engines become increasingly more and more technologically advanced so as to continually increase performance and efficiency, maintenance and repair of turbines have become increasingly more expensive. In an effort to control the increasing maintenance and repair cost, turbine users have pushed to increase the capabilities of turbine engine diagnostics. Artificial Intelligence (AI) techniques such as expert systems, artificial neural networks (ANN's) and fuzzy logic have helped make turbine diagnostics much more sophisticated, reliable, cost effective and easy to use. The current literature in turbine diagnostics predicts that future developments in artificial intelligence techniques will continue to make the turbine engine maintainer's job easier, faster and more accurate (Meher-Homji, 1985; Milne, 1987; Somers, 1990; Merrington, 1993; Kuo, 1995).

The United States Army Ordnance Center (USAOC) and the Army Research Laboratory (ARL) have developed a diagnostic system using artificial intelligence techniques for the M1 Abrams Main Battle Tank's AGT-1500 turbine engine known as Turbine Engine Diagnostics (TED). TED is a diagnostic expert system that assists the M1 Abrams mechanic. The system provides assistance during engine inspection and troubleshooting. It provides detailed information about the most frequently used maintenance procedures. It has an automated parts ordering system. Finally, it has a diagnostics tool capable of monitoring the engine's electronic signals.

1.1 Background

The AGT-1500 gas turbine engine (See Fig. 1) powers the world's most advanced tank - the M1 Abrams main battle tank of the U. S. Army. The AGT-1500 turbine engine is a variable geometry, free-power turbine engine utilizing a two spool compressor, single can combustor, a counterflow type stationary heat exchanger commonly known as a recuperator, an output reduction gearbox and accessory gearbox. For ease of maintenance and serviceability, the engine is composed of three basic modules: Forward Engine, Rear Engine and Accessory Gearbox. The modular design and small number of moving parts means more flexible logistics, fewer spare engines and lower life cycle costs (Textron Lycoming, 1990). The Forward Engine module includes the air inlet section, a low-pressure compressor, a high-pressure compressor, a combustor, a high-pressure turbine and a low-pressure turbine . The Rear Engine module includes a power turbine, a reduction gearbox and a recuperator. The Accessory Gearbox module includes a fuel management system consisting of an electro-mechanical metering device, a starter motor and an oil pump. The engine is equipped with a lubrication oil reservoir. The engine weighs 2500 pounds and its dimensions are 66 inches in length, 40 inches in width and 28 inches in height.

The engine delivers 1500 shaft horsepower (SHP). The power output of the AGT-1500 provides a power to weight ratio of 23.1 hp/ton. This is enough to accelerate the 65-ton M1 Abrams from 0 to 20 mph in 7.0 seconds. Top speed is governor limited at 41.5 mph. The turbine engine operates with remarkably low noise and vibration. The AGT-1500 has no smoke signature. It is smokeless even during acceleration and gear changes. The primary fuel is diesel but its true multifuel capability allows it to run on jet fuel, gasoline or even marine diesel fuel and leaded gasoline in emergencies. The

operating envelope includes a broad temperature range, -60^{O}F to 130^{O}F. (Textron Lycoming , 1990).

The AGT-1500 is unique among production gas turbines in that it uses a recuperator. This efficient stationary heat exchanger recovers most of the exhaust heat and transfers it to the compressed air before combustion. The recuperator improves fuel economy at partial power where a tank spends most of its time. By reducing exhaust heat, it also reduces the infared signature and exhaust noise (Textron Lycoming, 1990).

While the AGT-1500 has proven to be an extremely reliable engine, there are several frequently encountered problems. The most common problem is low power. This is actually a symptom that can be attributed to several different problems. Those problems include reduced air flow caused by dirty or poorly maintained air filtration system components. Low power can also be caused by failure of the electrical system or electrical failure of components such as the Electronic Control Unit (ECU) or the engine mounted sensors. Another problem that can cause low power is thermal degradation of the engine over long duration of use. This can be attributed to improper engine operation. The most common improper engine procedure is failure to perform the two minute cooldown at idle prior to engine shut off. Recuperator matrix cracking is another common problem. The most common cause of this is improper engine operation. Compressor surging caused by sand erosion is another common problem. Bearing failures is another problem with the engine. The primary cause of bearing failures is the result of "coke" build up in the oil passages. Bearing failures can also cause severe metal contamination of the lubrication system.

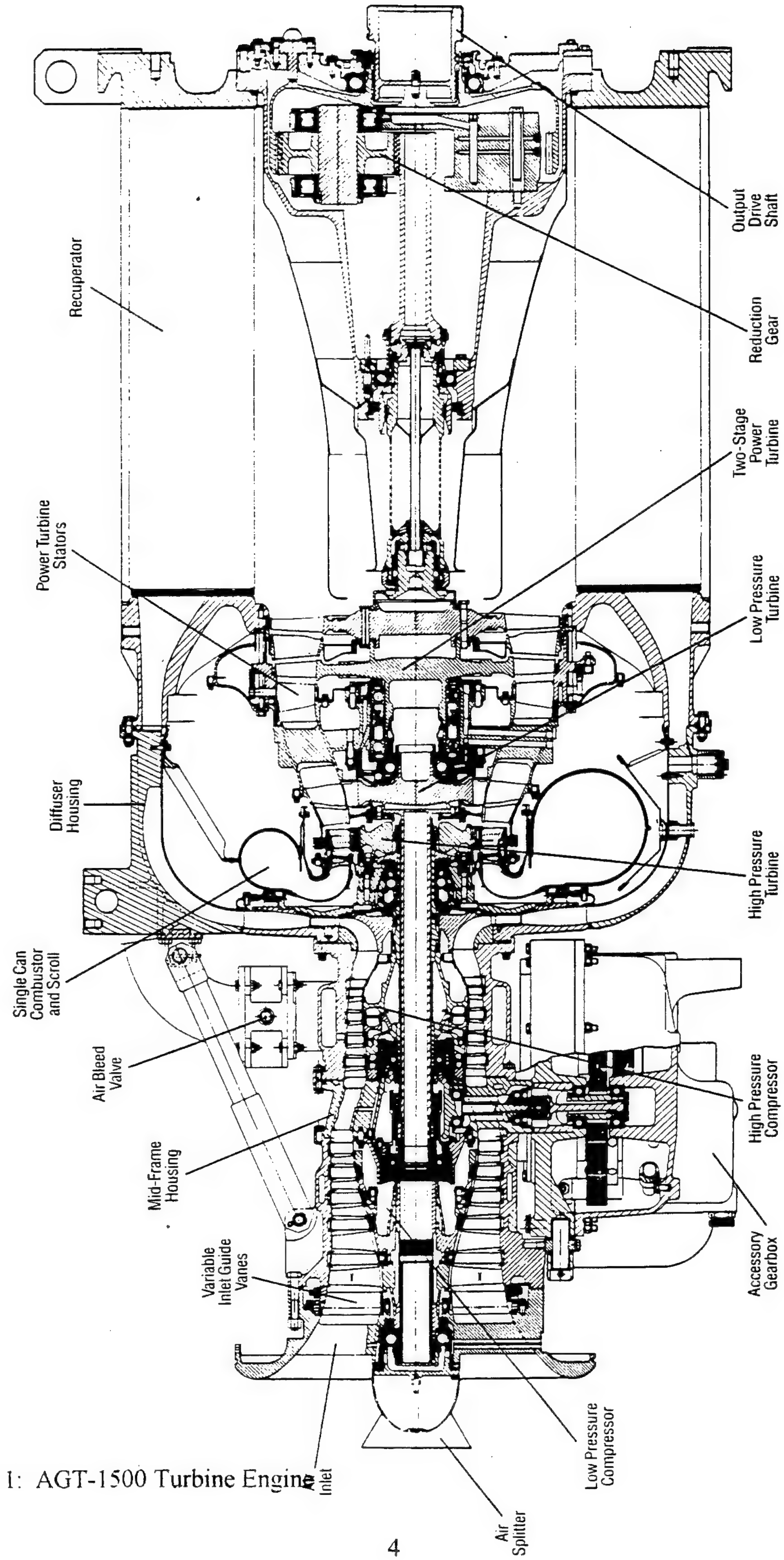

Figure 1: AGT-1500 Turbine Engine

In 1991 the Army decided that a better and more cost efficient maintenance system was needed for the M1 Abram's engine. the AGT-1500 gas turbine. The US Army Ordnance Center which is responsible for maintaining all Army ground vehicles, asked the Army Research Laboratory to work on a combined effort to improve the AGT-1500's turbine maintenance.

Many factors the led Army towards this decision to improve the AGT-1500's turbine maintenance system. The first and most important factor was the expense to operate, support and maintain the M1 Abrams. The Army has 7644 M1 Abrams in both active and reserve armor units as of September 1996. The AGT-1500 turbine engine accounts for the largest portion of the total M1 Abrams maintenance cost.

Another factor leading to the development of TED is the Army's maintenance doctrine. The Army has four levels of maintenance. The lowest level is organizational or unit level. The next level is direct support (DS). Above direct support is general support (GS). The highest level is depot. For the AGT-1500, there is no general support maintenance. So, engines that cannot be fixed at the direct support level have to be sent to depot at an approximate cost of $450,000 per engine.

It was decided that the TED program would be focused at the direct support level for several reasons. Army doctrine dictates that when an engine cannot be fixed at organizational level, it is pulled from the tank and sent to direct support. When the pulled engine arrives at direct support, another operational engine is quickly sent forward to return the tank to a fully operational status. During repair, direct support maintenance uses a ground hop support set (GHSS) to replicate the functions of the tank including fuel, battery, driver's instrument panel and electronic control unit. Also. under the Army's new "Fix Forward" maintenance concept, there is no general support maintenance

for the AGT-1500. The direct support units can now perform maintenance tasks previously only authorized at general or even depot level.

Finally, the M1 Abrams tank will be in the army inventory for the foreseeable future. The M1 Abrams first entered service in 1980. The original lifetime estimate was twenty years. However, the planned replacement vehicle project for the M1 Abrams was recently canceled causing the M1 Abrams service life to be extended to thirty years. That extension leaves armor units maintaining their M1 Abrams for at least the next 15 years and quite possibly longer.

1.2 TED Development

The approach of the TED programmers included some important guidelines for development of the program. In order for civilian programmers, with little or no prior military experience, to produce a product that the military mechanics had confidence in and were willing to use, they found it essential to learn the language of the subject matter experts (SME). To do this, the TED team observed the mechanics who would use the program in their own environment. They attended and videotaped classes for M1 Abrams mechanics. This resulted in three important benefits. It allowed the programmers to learn the language of the mechanic. It gave an accurate picture of how a mechanic performs his job. And it established a bond between programmer and mechanic.

The TED team also developed a first prototype quickly. The TED program started in 1991 and the first prototype was ready by January 1992. This rapid prototyping was essential for two-way communication. It allows the user the earliest

opportunity to comment on the system and gives him some idea as to the potential of the project.

Frequent testing was also an important part of the project development. Early in the project, the software was tested at least weekly in the USAOC. After the formal test in August 1993, testing was decreases to once a month.

At the beginning of the project in 1991, the Army had already chosen the hardware for the TED project. The Army planned to field the contact test set (CTS) III to its mechanics as part of the Army common hardware. CTS III is a DX-33 MHZ or DX-50 MHZ 80486 processor that employs 8 megahertz of RAM and either 200 or 500 MB hard disk drive. It is capable of running Unix, DOS or Windows. Windows was selected as the operating system because of its capabilities and perceived growth potential. TED was developed using the CTS speed and memory capabilities as constraints. However, as the project progressed, it became clear to the Army and the TED team that there was a problem with the CTS. The problem was reliability. The mean time between failures was only about one hour. This has caused the Army to stop fielding of the CTS. Instead, the Army will field TED to its active units with a Panasonic CF41 ruggedized laptop computer. Ruggedized laptop computers were developed for use in industrial situations. Initial testing by the TED team has shown that they hold up extremely well during use in Army maintenance environments. This provides not only better reliability but a much lower cost. The Panasonic laptops cost about $4,000 compared to over $15,000 for each CTS.

The TED team chose to use commercial off-the-shelf (COTS) software when available and suitable. When not available, they chose to wait for a new products to be released or upgraded or write the code in house.

The TED team quickly established several design goals. The software should be:

- accurate
- easy to use
- flexible
- task oriented
- should support several levels of expertise.

These goals were based primarily on the SME's extensive experience as M1 Abrams mechanics and as instructors for M1 Abrams engine maintenance classes.

1.3 Reasoning in TED

The main diagnostic software in TED is a Windows-based shell called Adept from Softshell. Adept is based on a reasoning paradigm called Procedural Reasoning System (PRS) which consists of a database, goals, plans, an intention structure and an interpreter.

The PRS database represents the current beliefs or facts of the system. These include facts about static properties, such as the structure of some subsystems or the physical laws that must be followed by certain mechanical components. Other beliefs are acquired by the PRS as it executes its plans. These can be current observations or conclusions derived by the system by observation (Georgeff and Ingrand, 1989).

Goals are represented by action descriptions and can be viewed as specifying a desired behavior of the system. This scheme of goals allows the PRS to express a much

wider class of goals including goals of maintenance, e.g. "achieve p while maintaining q true" and goals with resource constraints, e.g. "achieve p without using more than one tool" (Georgeff and Lansky, 1986).

Knowledge about how to accomplish given goals is represented in the PRS by plans or knowledge areas (KA's). Each KA consists of a body, which describes the steps of the procedure and an invocation condition, which specifies under what situations the KA is useful. The body of a KA can be viewed as a plan. It is represented as a graph with a start node and possible multiple end nodes. The invocation condition contains a triggering part describing the events that must occur for the KA to be executed (Georgeff and Lansky, 1986).

The intention structure contains all those tasks that the system has chosen for execution. These adopted tasks are called intentions. A single intention consists of some initial KA together with all the sub-KA's that are attempting to successfully execute the initial KA (Georgeff and Lansky, 1986).

The PRS interpreter runs the entire system. At any particular time, certain goals are active in the system and certain beliefs are held in the system database. Given these goals and beliefs, a subset of KA's in the system will be applicable. One or more of these applicable KA's will be chosen for execution and thus be placed in the intention structure (Georgeff and Lansky, 1986).

1.4 Contribution of Thesis

This thesis will investigate the TED project and its usefulness in the Army's maintenance of the AGT-1500 turbine engine. It is expected that the TED program will

help decrease the number of inconsistencies associated with human decision making and play an important role as a decision support system for the AGT-1500 mechanic.

Chapter 2

Artificial Intelligence

Before beginning a detailed description of TED, this chapter will provide a discussion of artificial intelligence (AI) and its practical applications, including descriptions of expert systems, artificial neural nets and fuzzy logic.

Artificial intelligence is concerned with the development of computer systems that have qualities we normally associate with human intelligence. These human intelligence qualities include the ability to learn from experience, to recognize patterns, to communicate, to solve problems, to adapt to new situations and many others (Ballast, 1989). Artificial intelligence is a science in relative infancy. The term artificial intelligence was only introduced 40 years ago. Like other sciences during their beginnings, progress has been difficult. This may be due to the nature of artificial intelligence. Ginsberg in his book, "Essentials of Artificial Intelligence" describes artificial intelligence as a mixture of long-standing problems from more established fields, including philosophy, linguistics, psychology, mathematics, physics, statistics, decision theory, biophysics and neuroscience combined with the power of the computer (Ginsberg, 1993). Even with all of the difficulties faced by people working in the area of artificial intelligence, progress is being made and the future looks good for more practical applications of this technology.

The following sections in this chapter will discuss three types of artificial intelligence systems: Expert Systems, Artificial Neural Networks (ANN's) and Fuzzy Logic.

2.1 Expert Systems

Expert systems are computer systems, consisting of both hardware and software. that use knowledge and inference or reasoning procedures to solve problems within a well-defined , narrow domain of knowledge normally handled by human experts (Ballast, 1989; Harvey, 1988; Liebowitz, 1995). Human experts are expected to solve problems, work with incomplete or missing information, draw conclusions based on probability rather than absolute certainty, explain results, determine relevance, restructure knowledge and learn by experience. A human expert also knows when to quit and accept the best possible answer. In order for a computer system to try to emulate these human expert characteristics, the expert system uses heuristics as well as standard algorithms. Algorithms are the fixed procedures that make up a standard computer program. Algorithms allow you to enter data and get precise and predicable results. Heuristics are most easily defined as "rules of thumb". They are the art of guessing in the absence of a reliable algorithm (Ballast, 1989).

Expert systems have many useful purposes. The foremost is to preserve or document knowledge so that a human expert's knowledge is not lost upon retirement or departure. Other important reasons include: a useful surrogate if expertise is unavailable, scarce or expensive, a training device, a vehicle for improving productivity, time and cost savings and a supplement or second check for the decision maker (Liebowitz, 1995). Expert systems consist of four major components: user interface,

inference engine, knowledge base and the knowledge acquisition system. Figure 2 shows the interactions of the major components schematically.

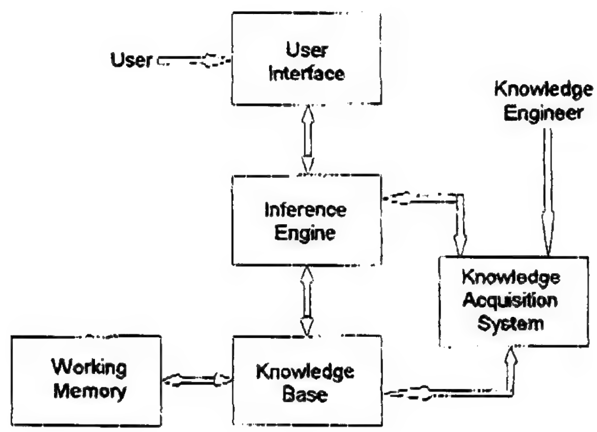

Figure 2: Major Components of Expert Systems

User interface or dialog structure allows the user to interact with the expert system. The user will present the problem to the expert system through the user interface. Then the computer will display or relay its conclusions back to the user through the interface. The hardware can be a video display terminal with alpha-numeric characters on a standard keyboard or a graphical user interface. As natural language capability becomes available, the interface between expert system and user should become more friendly. Eventually, speech recognition and generation may replace the keyboard and graphical interface (Myers, 1986). The more important aspect of the user interface is the ease with which the system can be used. The questioning by the system should be clear, data should be easy to input or even automatically relayed from the engine or data source and there should be a method for the system to explain why it is proceeding as it is (Ballast, 1989).

The inference engine or reasoning mechanism is similar to the control structure in a conventional computer program in that it operates deductively and selects relevant knowledge to reach a conclusion. The inference engine controls how the knowledge base and input from the user are manipulated to arrive at a solution. It decides where to start the reasoning process and the order in which inferences are made. Various inference techniques exist. Backward chaining or goal directed techniques start with a hypothetical fault and reason backwards to identify possible causes of the fault. Forward chaining or data driven techniques start with a list of requirements and then determine the conditions necessary to satisfy those requirements. In practice, many techniques will be required depending on the problem to be solved and the purpose of the knowledge system (Harvey, 1988; Ballast, 1989).

The knowledge base is perhaps the most important part of the expert system. It differs from an ordinary data base in that it contains not only a set of facts but includes heuristics or relationships in the expert system domain. The knowledge must be complete, consistent and accurate (Liebowitz, 1995). Facts include the information that we normally associate with a database: unchanging data and formulas that the system can use as needed. Relationships are the heuristic rules that represent the true knowledge of the expert system. There is also a working memory associated with the knowledge base. Here is where the inference engine manipulates user input with data from the knowledge base to arrive at a solution (Ballast, 1989). A major advantage of having the knowledge base separate from the inference engine is that knowledge can be added or changed without changing the inference engine (Harvey, 1988).

In order to charge the system with the facts, relationships and knowledge of human experts, there must be a way to modify the knowledge base and the inference engine. This is the function of the knowledge acquisition system. Once this is

developed, the acquisition system is usually invisible to the user and may not even be part of the working program. It is however, vital to the development and maintenance of the system (Ballast, 1989).

Expert systems offer many advantages. The most obvious is that expert systems capture human knowledge and judgment for use in areas where resources are scarce and therefore at a premium. Other advantages include the ability of expert systems to manage more complex systems, having information presented in a context dependent way, and having detailed reports and records maintained (Culp, 1989; Harvey,1988). Another recently developed advantage of expert systems is that they do not necessarily have to be built from scratch. There are "expert system shells" that can be used to help in the construction of the expert system. The expert system shells are typically the dialog structure and inference engine of the expert system. The expert system developer or knowledge engineer builds the knowledge base using the expert systems shell's knowledge base editor. In this manner, the knowledge engineer can concentrate on the development of the critically important knowledge base (Liebowitz, 1995).

The only obvious disadvantage of expert systems is the limited domain of each system. This problem may be overcome in the future through the integration of many individual systems (Ballast, 1989).

2.2 Artificial Neural Networks

Artificial Neural Networks have been defined many different ways. They are an artificial system motivated by the neural structure observed in real biological organisms (Moore, 1992). More specifically, they can be defined as a massively parallel interconnected network of simple, usually adaptive elements and their hierarchical

organization which are intended to interact with objects of the real world in the same way as biological nervous system do (Huang and Zhang, 1994; Momostori and Barschdorff, 1992). ANN's are also called parallel distributed processors, neurocomputing, connectionist models, neuromorphic systems and artificial neural systems. ANN's represent an attempt to simulate biological information processing through massively parallel, highly interconnected processing systems. ANN concepts date back to 1943 when McCulloch and Pitts suggested that computations could be performed by a network of simple binary neurons (Sondak, N. and Sondak V., 1989).

ANN's can be defined by three elements: a set of processing elements called neurons, a specific topology of weighted interconnections and a learning law which provides for updating the connection weights. A neuron is essentially a node with weighted inputs and a nonlinear function for an output.

ANN topologies fit broadly into two classes: feed-forward and recursive or feedback. Typical feed-forward networks have input layers, hidden layers and output layers. Each output in a layer is connected to each input in the next layer. There is no feedback between layers. Generally speaking, feed-forward networks are static. This means they produce only one set of output values rather than a sequence of values from a given input. Feed-forward networks are memory-less in the sense that their response to an input is independent of the previous network state (Jain, et al, 1996). In a recursive ANN, each neuron receives as input a weighted output from every other neuron in the net. Recursive networks are dynamic systems. When a new input is presented, the neuron outputs are computed. Because of the feedback paths, the inputs to each neuron are then modified, which leads the network to enter a new state (Jain, et al, 1996).

Learning laws are used to change the interconnection weights to some desired values used during an operational mode. Learning algorithms for weight adjustment can be described either as supervised learning, unsupervised learning or reinforcement learning (Moore, 1992). In supervised learning, the network is provided with a correct answer or output for every input. Weights are determined to allow the network to produce answers as close as possible to the known correct answers. Unsupervised learning does not require a correct answer associated with each input pattern in the training data set. It explores the underlying structure in the data or correlation between patterns in the data and organizes patterns into categories from these correlation. Reinforcement learning is a variant of supervised learning in which the network is provided with only a critique on the correctness of network outputs, not the correct answers themselves (Jain, et al, 1996).

ANN's provide the following main benefits:

- processing speed through massive parallelism;
- learning and adapting ability by means of efficient knowledge acquisition;
- robustness with respect to fabrication defects and different failures;
- compact processors for space and power constrained application

(Vankayala and Rao, 1993).

ANN's have several disadvantages:

- ANN's cannot explain its result explicitly, which implies that the user interface of ANN's may not be user friendly.
- The configuration of ANN's is vague and not easily understood.

- The current ANN learning algorithms are not efficient enough and cannot guarantee convergence.

- How to derive some type of optimal training set for ANN's still remains a question. (Huang and Zhang, 1994).

2.3 Fuzzy Logic

Fuzzy logic is broadly viewed as a system for dealing with reasoning that is approximate rather than exact. Fuzzy logic does not consider whether something is true or false, but how true a statement is. It allows for shades of gray between absolute truth and falseness (Klir, 1995; Schofield, 1995; Shandle, 1994).

Fuzzy logic is based on fuzzy set theory, a mathematical discipline invented in 1965. Unlike the classical notion of a set, the boundary of a fuzzy set is not required to be precise. Membership in a set is not a matter of affirmation or denial, but a matter of degree. In fuzzy logic, a statement such as "Mark is driving fast" can be 75% true and 25% false if Mark is driving at 70 mph. Functions that compare and classify input sensors data for the purpose of defining the fuzzy set are called membership functions. Fuzzy logic lets engineers control real world systems by using language based rules rather than rigorous mathematical modeling. It has proven especially effective for situations having complex equations, lots of exceptions to the equations, nonlinearities to accommodate, system inputs that provide vague or ambiguous information or no equations at all (Klir, 1995; Schofield, 1995; Shandle, 1994).

A significant advantage of fuzzy logic is a shorter development cycle. Typical conventional system solutions have five development steps: developing a model for the system, developing a mathematical model for the controller, analyzing the equations,

converting the model to a circuit and building the circuit. Fuzzy logic on the other hand has only three: developing the fuzzy model, then simulating it and compiling it (Shandle, 1994).

A disadvantage of fuzzy logic is the complexity. As the number of inputs increases, the number of rules and membership functions increase exponentially. Developing the rules and membership functions can be extremely time consuming with no way to verify the resulting system other than trial and error.

In cases where developing the rules and membership functions is difficult, ANN's make an excellent companion. Generating the rules and membership functions is essentially a learning process. Learning is a strength of ANN's. ANNs use their self learning ability to generate the rules and membership functions for a fuzzy logic based controller. Using supervised learning, an ANN can sort out the rules and membership functions that make a system behave as the designers wish (Khan and Rahman, 1995; Shandle, 1994).

Chapter 3

Turbine Engine Diagnostics (TED)

The entire TED system includes three main modules and two special applications (See Fig. 3). The first module, entitled TED, directs the mechanic to the bulk of the diagnostic and maintenance expertise. The second main module, automated breakout box (ABOB), allows the automatic interrogation of the signals from the engine. The last main module, repair parts and special tools list (RPSTL), contains the automation of the repair parts and special tools list. The two special applications are the diagnostic intelligent tutoring system (DITS) and special system administration functions.

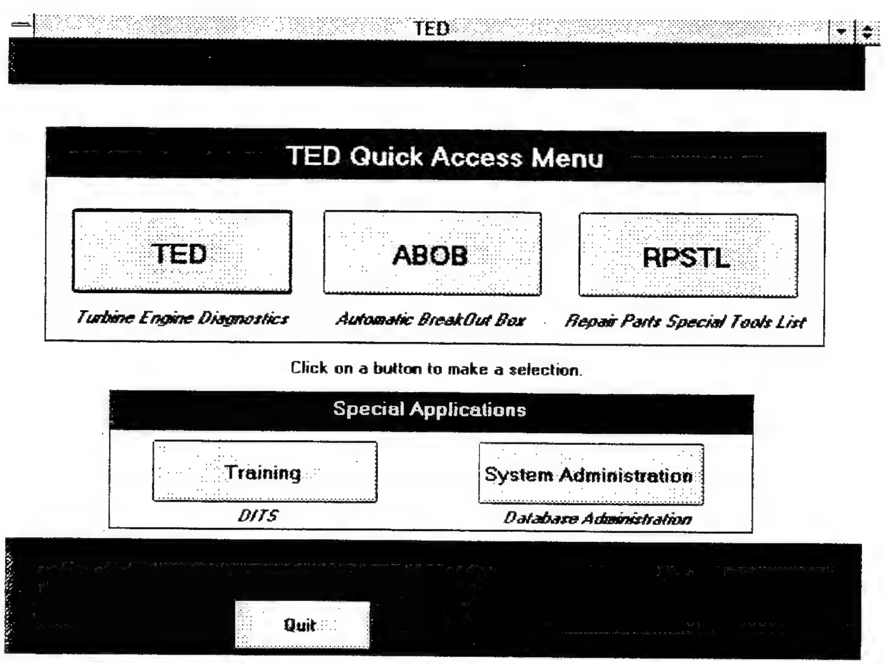

Figure 3: TED Main Menu

3.1 TED Module

The TED module (See Fig. 4) is divided into three main troubleshooting and maintenance sections: initial and final inspections, operational checks and maintenance. The user also has direct access to the RPSTL module from the TED module menu.

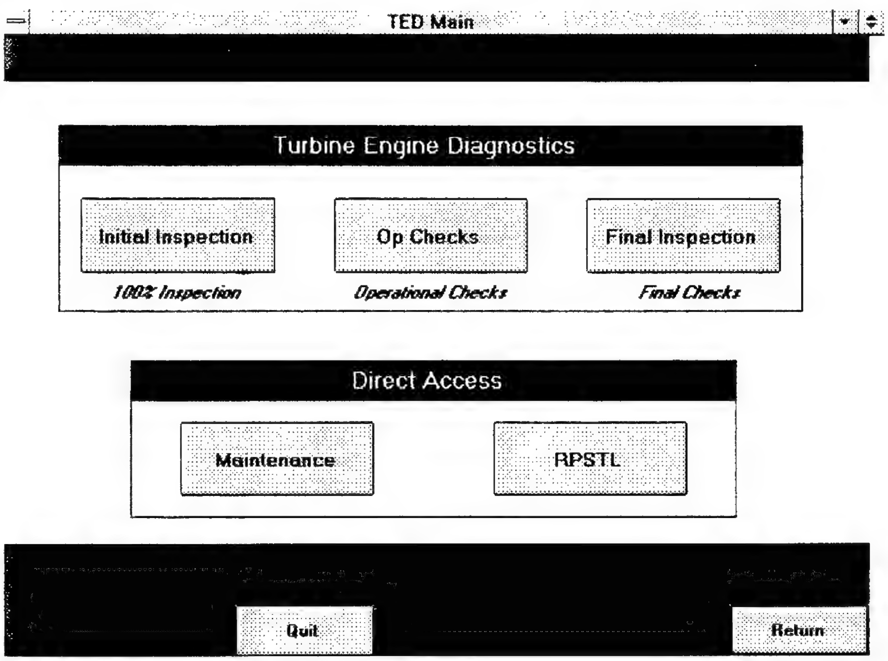

Figure 4: TED Module Main Menu

3.1.1 Inspections

The initial inspection routine guides the mechanic through a complete inspection of the AGT-1500 engine. Each "page" of the routine allows for many levels of expertise (See Fig. 5). The experienced mechanic can just look at the list of items to be inspected, note any faults on the electronic DA Form 2404, Equipment Inspection and Maintenance

Worksheet, then quickly move on to the next page. The less experienced mechanic has several options to assist him during the inspection. First. each page has an actual photograph of the section of the engine where the items listed for inspection are located. If the mechanic is unsure exactly where on the engine the picture is depicting, the "Location" button can help. The "Location" button displays an overhead schematic view of the engine with a red dot indicating the perspective from which the picture was taken. Simply clicking on each item in the list to be inspected gives the mechanic a more detailed picture of the item with a red line pointing to its exact location. This will also give a list of the possible deficiencies that the mechanic should look for on the item. Clicking on the "How" button will give the mechanic the same information but will display the information sequentially for each of the items on the inspection list for that page. The "Why" button is something new for Army maintenance manuals. The "Why" button tells the mechanic the reason for the inspection and the possible consequences of deficiencies. The "Why" button will also display safety warnings for the mechanics. Each page gives the mechanic access to a notepad to write reminders or information not directly related to a deficiency. Finally, each page allows the mechanic to view all the listed faults on the DA Form 2404 or quit the inspection. Quitting will generate a partial inspection sheet. The final inspection routine is currently not available.

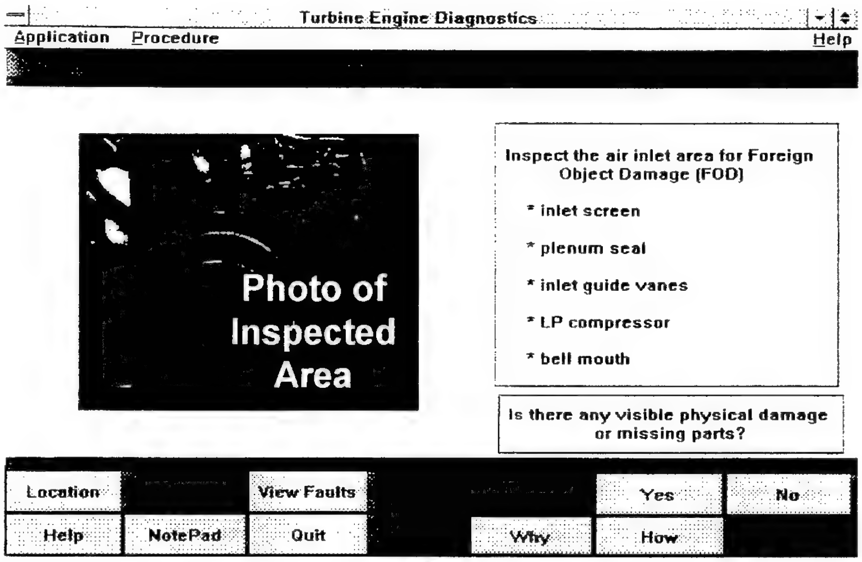

Figure 5: Typical Initial Inspection page.

3.1.2 Operational Checks

The operational checks routine offers troubleshooting diagnostic procedures in six areas: Protective Modes, No Start, Low Power, Rapid Functional Assessment (RFA), High Oil Consumption and Quick Coastdown (See Fig. 6). Each of the six sub-modules contains diagnostic logic to first determine the cause of the faulty symptom. Once the cause has been detected, TED will link to the appropriate maintenance and repair parts manual. The routines again offer step by step instructions and any necessary safety warnings for all the troubleshooting. Each "page" includes photographs of the troubleshooted area and when appropriate the "How" and "Why" buttons.

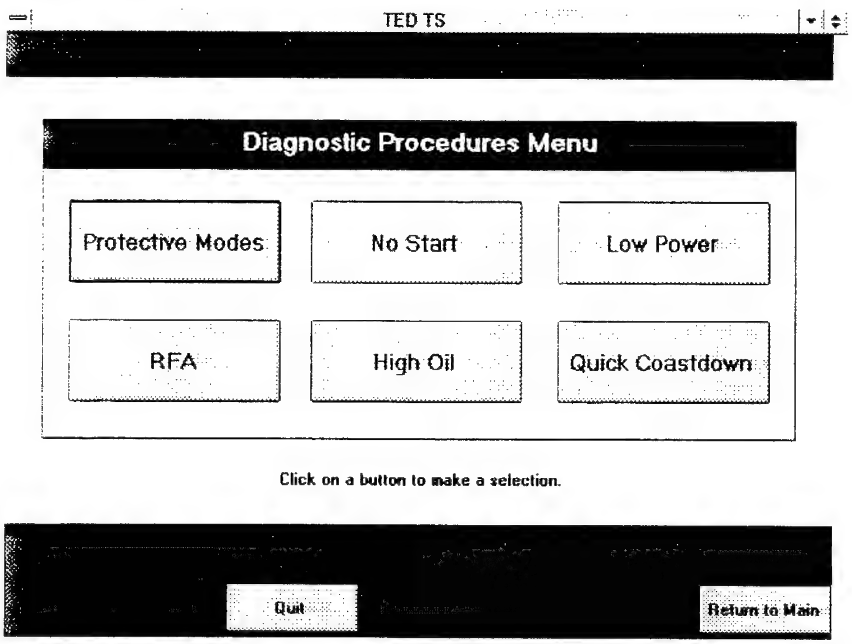

Figure 6: Operational Checks Menu.

-Protective Modes. The Electronic Control Unit, which constantly monitors all sensor inputs and compares them with established parameters, can initiate one of four protective modes to prevent damage to the engine. However, at the DS maintenance level, there was previously no way to query the ECU to find what protective mode was invoked. This module will first check to see if a protective mode condition exists then identify the cause and link to the appropriate maintenance and repair parts module.

-No Start. Using a few basic questions as to why the engine will not start, this module quickly and accurately determines the most probable cause of a no start condition.

-Low Power. This module is designed to isolate faults and repair a low power condition found by the engine health check.

- Rapid Functional Assessment (RFA). This module quickly determines whether it is safe to attempt to start the engine. A minimal number of inspections, three rotational and four lubricational, evaluate the mechanical integrity of the engine's internal rotating components. If problems are found, RFA will isolate the fault or for more serious problems, recommend replacement of the engine.

- High Oil Consumption and Quick Coastdown. These modules are currently not available. The TED team expect to include them in the next published version of the program.

3.1.3 Maintenance

This module contains maintenance procedures for six sub-systems of the AGT-1500 engine: Engine, Engine/Transmission, Full up Power Pack (FUPP) Charging System, Smoke Generator and Cooling System (See Fig. 7). Procedures include adjust, repair, remove and replace the components of each sub-system. The mechanic can use the routines in the browse or data-driven mode. When in the browse mode, the mechanic selects maintenance procedures manually through menus and sub-menus. This provides the experienced mechanic the flexibility of viewing only procedures that they need while bypassing familiar or routine tasks. In the data-driven mode, TED automatically establishes the correct links to all pertinent maintenance procedures and to appropriate sections of the RPSTL module.

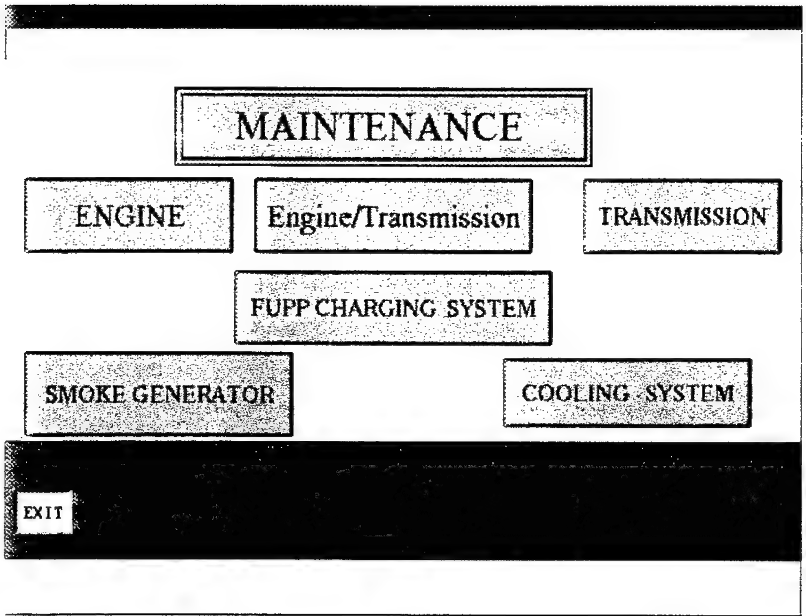

Figure 7: Maintenance Procedures Menu.

3.2 Automated Breakout Box (ABOB)

The module entitled ABOB of the TED program allows the mechanic to interface with the diagnostic tool, ABOB. Mechanics utilize the ABOB connected to the M1 Abrams Electronic Control Unit. The use of the ABOB is an alternate troubleshooting method to determine the operational status of the AGT-1500. The ABOB is a firmware program capable of converting 128 separate analog signals into digital format. The ECU's J1 diagnostic connector provides 32 analog signals to the ABOB. The ABOB contains a 128 to 1 multiplexer and an analog to digital converter, both operated by an 8-bit embedded controller (Kangas et al, March 1994). Once the ECU analog readings are converted into a digital format, these signals are passed to the TED program through a standard serial port. When the ABOB module of the TED program is run with the ABOB, signals can be automatically monitored and when a fault occurs, the mechanic is notified of the problem.

The ABOB main menu screen contains seven buttons: Show/Store, Turret, Fault Finder, Check Adjust, ABOB Help, Test ABOB, and Quit (See Fig. 8).

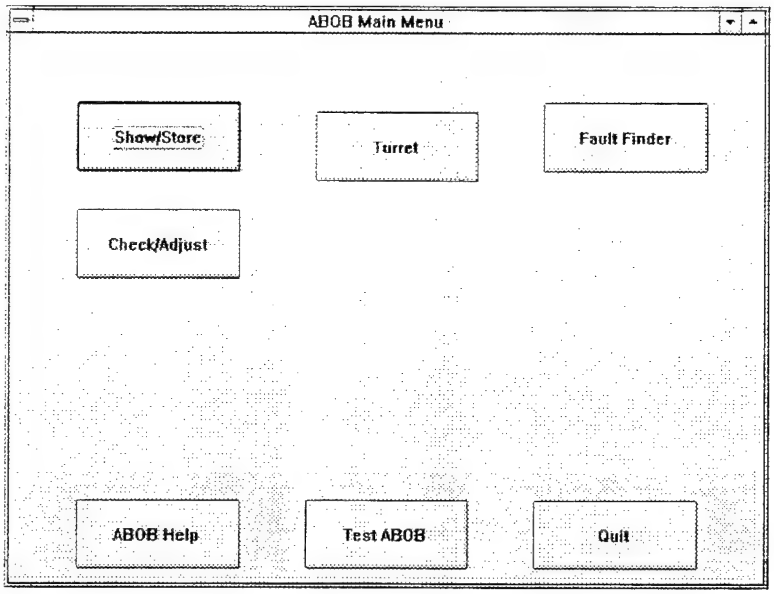

Figure 8: ABOB Main Menu.

-Show Store (See Fig. 9). This button is used to display up to 32 pins on the screen and to store them for later viewing and analysis. The top part of the Show/Store screen lists the 32 pins. The mechanic may select any number of pins to monitor. The standard default is 21 selected pins. The middle of the screen allows the mechanic to choose to display raw voltages or physical units and either save the information to disk or display on the screen. The bottom of the screen shows the output file name and file header if the save to disk option is selected. The "Run" button on the screen will use the choices selected and show voltages to the screen or save to disk. The ABOB will continue to monitor voltages until the "Stop" button is selected.

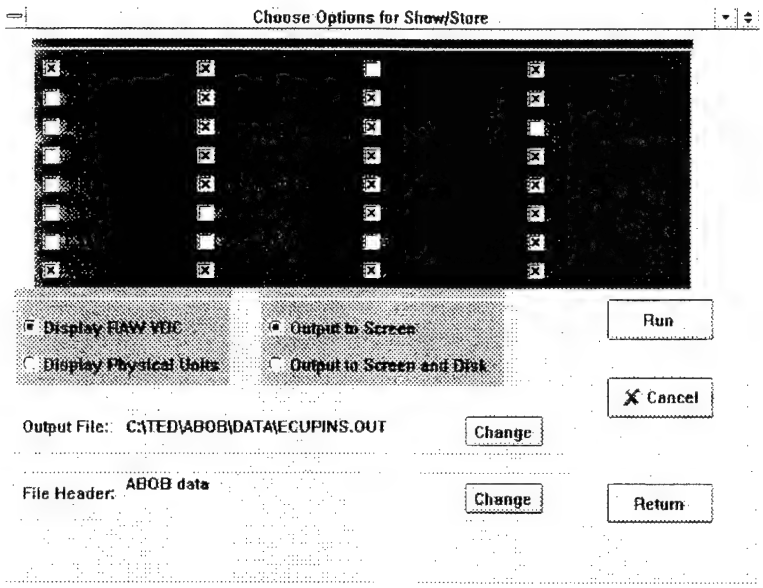

Figure 9: ABOB Show/Store Screen.

-Turret. This button will eventually run a program to view the signals from the Turret Network Box J1 connector (TNB-J1) . This button was included in the ABOB module to show that the ABOB was capable of reading signals from other than the ECU-J1 connector. There is currently no diagnostic logic in this module.

-Fault Finder (See Fig. 10). This button runs a program to determine the cause of a "No Start" or a "Protective Mode" condition. The top of the Fault Finder screen monitors the six available engine indicators: High Pressure Compressor Speed (NH), Turbine Speed (NPT), Ambient Inlet Air Temperature (T1), Fuel Flow Actual (WF Act) and Fuel Flow Requested (WF Req). To the right of this are two warning indicator lights: Weight of Fuel Fault Mode (WF FM) and Gas Over Temp. The WF FM warning light is triggered by either a low fuel flow or by an error between the requested fuel flow (WFR) and the actual fuel flow (WFA). The Gas Over Temp light is triggered when the temperature at the T7 (the 7th bearing assembly) averaging harness exceeds the designated threshold. Below these are three boxes with three indicators each. The left and middle box show

the engine condition. The right box indicates problems with the ABOB or ECU. Below these boxes is a box with four indicators that show whether the components are open or closed, powered or not powered. To the right of this box is a rectangle that displays a clock when a Start Attempt is detected. The clock will display the elapsed time from beginning of a start attempt then elapsed time from a successful start and finally the elapsed time from a shutdown. Below this is a box containing the four Protective Mode indicators. These will light to indicate that a protective mode has been found. Below this is a list of twelve possible causes for protective modes. ABOB will attempt to find all possible causes for the fault and light the appropriate fault indicator. Below the clock is the Show Fault Diagram button. This button will display a chart of all the protective modes found and their possible causes.

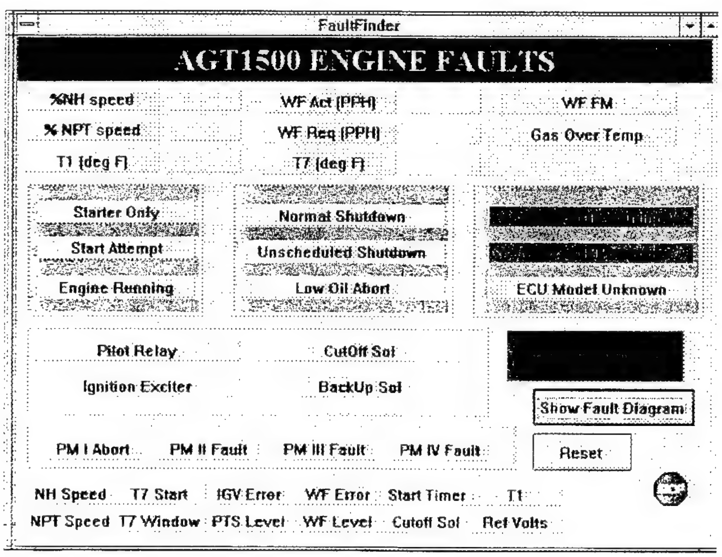

Figure 10: ABOB Fault Finder Screen.

-Check/Adjust. This button is used for engine static checks, engine health check and engine adjustment checks. The engine adjustment checks include the inlet guide vane check, power lever angle check and power turbine stator check.

-ABOB Help. This button starts a tutorial showing how to hook up the ABOB.

-Test ABOB. This button runs a self test on the ABOB and reports the results.

-Quit. This button will shut down the ABOB program and return to the TED main menu.

3.3 Repair Parts and Special Tool List (RPSTL)

The RPSTL module allows the mechanic to easily access the parts ordering information for every aspect of the M1 Abrams engine and transmission. The mechanic may be automatically linked to the RPSTL from a diagnostic procedure as appropriate. He can also use the module independently from a table of contents or choose to search for a specific part using the part number, national stock number or nomenclature. All required information is provided so the mechanic can automatically order parts and file parts requests.

The RPSTL main menu has three sub modules: Technical Manual (TM) Lookup, Data Lookup and Administration (See Fig. 11).

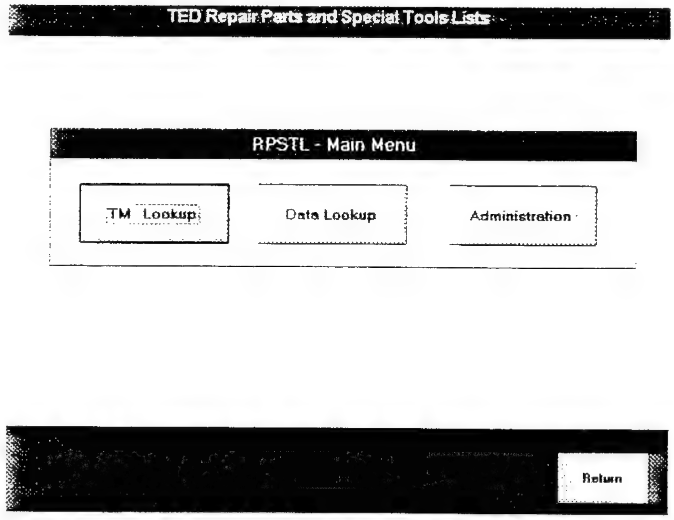

Figure 11: RPSTL Main Menu.

The first two sub modules, TM and Data Lookup are search methods. The TM lookup button permits the user a fast method to display and then interrogate selected subsystems from either of the major subsystems: engine, transmission or accessories (accessories currently not available). Clicking on one of the buttons will display the subsystems of the major subsystem. Once the mechanic selects one of these subsystems, a list of figures for that subsystem is displayed. For each figure, its associated parts list is displayed on the right side, while its drawing is detailed on the left (See Fig. 12). Items are selected from the parts list by clicking on the particular order box. The detailed drawing has the capability to be magnified so that particular areas of interest are easier to see.

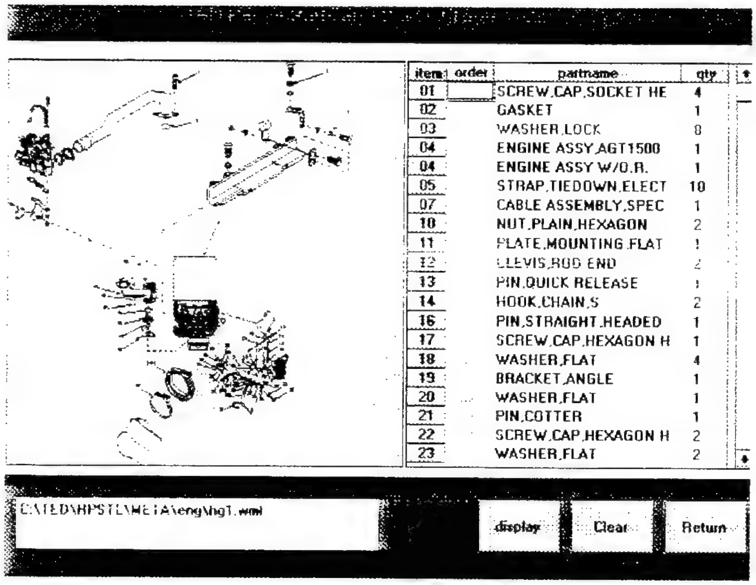

Figure 12: Typical RPSTL Display.

The Data Lookup button searches all of the major subsystems using part number, national stock number or nomenclature. If the search is successful, a list of figures containing the item will be displayed. Parts then may be ordered in the same manner as above.

The Administration submodule supports general report and database upkeeping routines.

3.4 Diagnostic Intelligent Tutoring System (DITS)

The Diagnostic Intelligent Tutoring System (DITS) is a special application of the TED program. DITS is a tutorial system that includes basic maintenance procedures, theory of engine operations and guidance on some important general maintenance procedures. DITS has two goals. First, it lets the mechanic review information to

expertly troubleshoot the M1 Abram's AGT-1500 engine. Secondly, it allows the mechanic to practice performing troubleshooting tasks.

DITS is adaptive. It will determine the mechanics level of experience with troubleshooting the engine, his troubleshooting skills and related knowledge and the ways in which he prefers to learn information. DITS will select or help the mechanic select the information to review and the procedures to practice.

DITS contains the following :

-An introduction which all new students must go through. The introduction explains what DITS is and how to use it.

-A short set of questions which new users must answer. The questions are used by DITS to determine the users skill level and preference for type of presentations.

-Four review modules which cover different aspects of the AGT-1500 engine and troubleshooting. This material is available to the mechanic before or after the hands-on troubleshooting practices.

-Some troubleshooting procedures for the mechanic to practice his skills at identifying faults in the AGT-1500 engine.

DITS also comes with some additional aids. There is a built in coach called "Sergeant". This coach will introduce modules and give advice during troubleshooting practices. In addition to the coach, there are the following student aids available from the top menu bar:

-a glossary of terms

-a notepad

-DITS Help which gives information about how to use the different features in DITS.

-a bookmarking feature which allows the mechanic to mark a screen for later reference.

-a tutorial on how to use MS-windows and a mouse.

3.5 System Administration

Systems Administration is the second of TED's special applications. The module contains the programs report writing and database maintenance functions. The routine allows the mechanic to access information associated with a work order. The work order information may be displayed, printed, archived or deleted.

Chapter 4

Field Testing and User Training

The initial field test of the TED software was during the week of 15 - 24 August 1993 at Fort Stewart, Georgia. The test soldiers were from the Support Squadron/278th Armored Cavalry Regiment (ACR) and the 771st Maintenance Company/176th Maintenance Battalion of the Tennessee National Guard (TNNG). The 30 soldiers that participated in the test had a wide range of experience. In addition to the fact that the soldiers equally represented the three Army maintenance skill levels (10 at each level), the TNNG was just transitioning to the M1 Abrams and therefore had very little prior experience with the AGT 1500 engine.

4.1 Test Objectives and Configuration

The test had two objectives. The first was to measure how accurately and quickly the mechanics could identify randomly assigned faults on the engine using TED versus the paper technical manual (TM). Secondly, the test was to determine if the program was soldier friendly and easy to use (Helfman, et al, 1994).

The test configuration consisted of four AGT-1500 engines that were removed from the M1 Abrams. This simulated the environment that the direct support mechanics would normally work on the engine. At two of the engines, TED software was provided for the mechanic to use. On the other two engines, current paper technical manuals were provided for the mechanics.

On each of the engines, several faults that would normally require attention by a mechanic were installed. The number of faults installed on an engine ranged between two and four and were randomly determined prior to the test. New faults were randomly assigned at the beginning of each day to avoid any carryover of information.

4.2 Conduct of the Test

The mechanics were randomly assigned to one of the experiment stations. Then they were instructed to conduct a complete visual inspection of the engine and complete a DA Form 2404, Equipment Inspection and Maintenance Worksheet. The test was complete when the mechanic finished the inspection process or when one hour elapsed. Each station was assigned an observer who recorded the time history of the inspection. The observers noted the start time, the time that each fault was detected by the mechanic and the end time. The observer was not allowed to interact with or coach the mechanic in any way.

4.3 Test Results

The test results show a very definite trend. At each skill level, mechanics using the TED software to conduct the inspection were more successful than those using the technical manuals procedures. The following is a comparison of the two methods.

Skill Level	Faults Detected Using Technical Manuals	Faults Detected Using TED Software
1	26%	52%
2	11%	42%
3	42%	56%

Table 1: Field Test Results

TED assisted the mechanics in finding at least twice as many faults as compared to TM's.

Also of interest is the mean length of time spent on each inspection. The mean time spent by a mechanic with TED was 49 minutes compared with only 20 minutes with the TM's. This longer time may help explain the superiority of TED over the TM in terms of fault detection. The TED software directs both the order of inspection and the items inspected. The TM does not structure the inspection process. TED seems to encourage the mechanics to spend more time on solving the problem. (Taylor and Monyak, 1994).

The ease of use was also readily apparent. Soldier acceptance was unanimously positive. Mechanics with both limited and extensive use with computers readily accepted TED as the preferred tool for maintaining the AGT 1500 engine. (Helfman et al, 1994).

4.4 User Training

In July 1994, fielding of the TED software began to the National Guard. Training for the fielded units began in January 1995 and continued through September 1995. Training was conducted by the ARL, instructors for the Ordnance School and by contractors from the TED team. Five "train the trainer" sessions were conducted, each 2-3 days long, both at unit locations and at the Ordnance School at Aberdeen Proving Ground, MD. All sixty five fielded units from twenty nine states sent a representative to attended the training.

Chapter 5

Turbine Engine Diagnostics Artificial Neural Network (TEDANN)

In 1994, The Pacific Northwest National Laboratory with the U. S. Army Ordnance Center developed a system that employed an Artificial Neural Network (ANN) to perform diagnosis and prognosis of fuel flow problems in the AGT-1500 gas turbine engine of the M1 Abrams. ANN technology was chosen for the prototype because it is well suited for diagnostics in real world applications. Each turbine engine is unique in its behavior as a result of its age, manufacturing tolerances and the environment in which it is operated. A diagnostic system using ANN's can automatically adapt to these individual variables for each engine (Illi et al, 1994). The initial focus of the system was to diagnose three faults in the main metering fuel valve during the turbine engine startup sequence: bouncing valve, sticking valve or stuck valve.

The ANN diagnoses faults by monitoring signals from the ECU which represents the status of the Electro-Mechanical Fuel System (EMFS). The EMFS is a fuel metering device that delivers fuel to the turbine engine under the management of the ECU. The signals that determine the fuel flow are throttle position, ambient air and power turbine inlet temperatures and compressor and power turbine speeds. Each of these variables has a representative voltage signal available at the ECU's J1 diagnostic connector, which is accessed via the ABOB.

The ANN's for TEDANN were trained using sensor data collected from turbine starts by Textron (the turbine engine manufacturer) and by the U. S. Army Ordnance

Center and School. Initial data sets were collected from mostly fault-free starts. The data was analyzed to understand how the sensor values behave during fault conditions. TEDANN analyzes the sensor data in the form of "features" computed from the date. During development these features were recognized as discriminators among the three fault conditions. However, use of sensor values as the one input to a simple feed-forward ANN do not capture information in the time domain. Thus, to capture time dependent information, input to the ANN included first derivatives of sensor values and first derivatives of differences between pairs of sensor values (Kangas et al, 1994).

Analysis by the ANN determines which fuel flow fault readings are out of tolerance with the EMFS nominal operational parameters. TEDANN will display either the fault status message identifying the EMFS faults or a message stating that the EMFS is fully operational.

The team that developed TEDANN proposed that it should be integrated with TED. When the integrated with TED, the output of TEDANN would have been submitted to the TED expert system for further processing. This integration would have provided a more realistic interpretation of the fault values using rule-based post processing of TEDANN's output (Kangas et al, 1994). The TED team is having a difficult time with this integration. In order to test the module, the fuel flow faults that TEDANN is designed to detect must be installed on an engine. The TED team currently has no way to install those types of faults on a test engine.

Chapter 6

TED's Future

The current success of the TED project has led to discussions about expansion of the project into many other areas.

The first area of possible expansion is to create a similar program for other vehicles in the Army's inventory. First among those mentioned as possible candidates is the Army's Bradley Fighting Vehicle (BFV). Like the Abrams, the Bradley is a battle proven armored vehicle that is expected to stay in the Army's inventory for a very long time. The benefits in cost savings and mechanic proficiency seen in the TED project could easily translate to the Bradley with a similar program. The TED team has had initial discussions with the managers of the Bradley's maintenance program and expect that the discussions will translate into a project in the next few years.

Other vehicles that could also benefit from a program similar to TED are the Army's huge inventory of diesel trucks. While the cost of maintenance per truck is nowhere near the cost of maintenance per tank, the cost savings could be significant if a program like TED was introduced for diesel trucks. Now would be a critical time to get started as the Army is currently in a program to replace most of it's current inventory with the new Family of Medium Tactical Vehicles (FMTV).

Other expansion efforts could be made into the Army's inventory of helicopters. The increased number of sensors on a helicopter would certainly make the

diagnostic program simpler to develop and possibly more accurate. Some advances have already been made in this area. The Army has tested a data recording system known as the Portable Engine Analyzer Test Set (PEATS) which is used to detect and diagnose faults in helicopter turbine engines. PEATS allows electronic data recording and transfer into a personal computer, where artificial intelligence methods are used to form and interpret the data (Pettigrew, 1995).

Another direction for possible TED expansion is into other systems on the M1 Abrams. The AGT-1500 turbine engine maintenance cost is only part of the total maintenance cost for the M1 Abrams. Other systems that are candidates for a TED type program include the turret and the hull of the M1.

The possibility of creating a version of TED for the organizational level mechanic is yet another direction that the TED team is exploring. Providing a similar program designed for the next lower level of mechanic could help realize even greater cost savings without a significant development time or cost.

Finally, the TED team is looking into other areas of diagnostic logic, including fuzzy logic and artificial neural networks, to increase the diagnostic capability of the TED program.

Chapter 7

Gas Turbine Diagnostic's Future

With the success of many new diagnostic systems and techniques, it is clear that the field will continue to grow and expand. This chapter will discuss some of the possible areas of future expansion for gas turbine diagnostics. The discussion will include the types and characteristics of future systems and the possible branching into the closely related area of prognostics.

Currently, most diagnostic systems are very specific. A system is designed for a specific turbine engine or a closely related family of engines. As the technology develops, the possibility of creating a more generalized or generic diagnostic tool or system may become feasible. The generalized system would work on large classes of engines. This ability to use a single diagnostic system on many different types of engines would realize immediate and significant cost savings. The difficulties of creating such a generalized tool, however, are quite obvious. Turbine engines come in many different sizes, configurations and types. Available sensor data is often different for each engine. And engine data and tolerances vary widely even for engine of similar configurations. All of the above are significant obstacles to creating a generalized diagnostic system.

An area that is ongoing and will certainly continue into the future is the retrofit of diagnostic systems onto older turbine engines. While it may be easiest and most efficient to design a diagnostic system at the same time as the engine is being designed, the cost savings of designing diagnostic systems for older engines are clear. The difficulties in

this area include the difficulty of accessing sensor data and retrofitting of any required hardware on the turbine engine.

Linking geographically separated diagnostic tools into a larger information source also has interesting possibilities. The information sharing capabilities of the World Wide Web (WWW) and the Internet could provide diagnostic systems with information on replacement parts availability, sensor data trends in similar engines, and a significant amount of other useful information. It could provide the mechanic with advice and guidance from other mechanics that have experienced similar problems.

Another area of expansion is the "hands free" operation. This type of system would include some type of wearable computer with a monitor and voice activated software. McDonnell Douglas is one of several companies developing these types of systems. The McDonnell Douglas system is called the Maintenance And Repair Support System (MARSS). MARSS is a body worn computer with a head mounted video display, stereo headphones and microphone that guides the mechanic through maintenance tasks. Voice activated software and wireless interface allows for untethered mobility and head-up-hands-free operation. (McDonnell Douglas, 1996).

A closely related area that warrants discussion is prognostics. While diagnostics tells you about the health of an engine, prognostics attempts to predict when an engine or engine part will fail. Current generation diagnostic systems are designed to identify individual events or trends in the output of sensors mounted on the engine. The system may provide a useful indication that a failure condition may be developing, but it cannot provide reliable predictions of the remaining safe or operational life (Hansen, et al, 1995). While the cost savings of an accurate prognostic system are obvious, the creation of such a system has proven to be very difficult. The lack of a significant amount of

literature in the area of prognostics is evidence of this difficulty. The reasons for this difficulty are many. Prognostics systems would require a large history of the timetables for mechanical failure. This type of historical data is often difficult and expensive to produce. Turbine engine parts often suffer from catastrophic mechanical failures. These types of failures rarely present any clues that they are about to occur. While other mechanical failures may present some type of vibrational indication of failure, most engines are not currently equipped with sensors able to detect such clues. Electronic parts rarely offer any type of indication they are about to fail.

While the outlook may currently not look good for prognostics, there are some areas that offer some cause for hope. As mentioned earlier, vibrational changes are often indicators of possible failure. The addition of sensors designed to measure vibration data and the collection of a large pool of normal and abnormal vibration history may offer some indication of time to failure. Additional chemical sensors in the engine's lubricational fluid that continuously look for anomalies, such as iron particles, may provide some insight into the time to failure.

Until future breakthroughs allow prognostic systems to become more feasible, the use of good diagnostic equipment and an aggressive preventive maintenance program will provide a cost efficient substitute for prognostics.

Chapter 8

Conclusions

The outlook for the future of turbine engine diagnostics seems favorable. Continuing breakthroughs in the artificial intelligence techniques discussed, including expert systems, artificial neural networks and fuzzy logic, should help provide the turbine engine maintainers with easier and more cost effective diagnostic tools.

The outlook for the TED project seems equally favorable. The TED team's bottom up approach to developing the program appears to have many benefits. The two way communication between the user and the programmer, the rapid prototyping and the frequent testing of software have allowed the team to create a tool that the mechanics not only find easy to use but appears to double the chances of detecting a fault. The success of the TED program will depend entirely upon the soldiers that use it. Their acceptance of TED as a diagnostic tool to assist them with maintenance is critical to realizing the potential cost savings.

List of Acronyms

ABOB - Automated Breakout Box

ACR - Armored Cavalry Regiment

AI - Artificial Intelligence

ANN - Artificial Neural Network

ARL - Army Research Laboratory

BFV - Bradley Fighting Vehicle

COTS - Commercial Off the Shelf

CTS - Contact Test Set

DITS - Diagnostic Intelligent Tutoring System

DS - Direct Support

ECU - Electronic Control Unit

EMFS - Electro-Magnetic Fuel System

FMTV - Family of Medium Tactical Vehicles

FUPP - Full Up Power Pack

GHSS - Ground Hop Support Set

KA - Knowledge Area

MARSS - Maintenance and Repair Support System

MHZ - Megahertz

NH - High Pressure Compressor Speed

NPT - Power Turbine Speed

PRS - Procedural Reasoning System

PEATS - Portable Engine Analyzer Test Set

RFA - Rapid Functional Assessment

RPSTL - Repair Parts and Special Tools List

SME - Subject Matter Expert

T1 - Ambient Inlet Air Temperature

T7 - Power Turbine Inlet Temperature

TED - Turbine Engine Diagnostics

TEDANN - Turbine Engine Diagnostics Artificial Neural Network

TM - Technical Manual

TNB - Turret Network Box

TNNG - Tennessee National Guard

USAOC - U. S. Army Ordnance Center

WF Act - Weighted Fuel Flow Actual

WF FM - Weighted Fuel Fault Mode

WF Req - Weighted Fuel Flow Requested

WWW - World Wide Web

References

Ballast, Kent D., "An Introduction to Expert Systems", The Construction Specifier, V 42, n 1, Jan 89, pp 88-93

Culp, Charles H., "Expert Systems in Preventive Maintenance and Diagnostics", ASHRAE Journal, V 31, n 8, Aug 89, pp 24-27

Georgeff, Michael P. and Ingrand, Francois, F., "Decision-Making in an Embedded Reasoning System", Proceedings of the 11th International Joint Conference on AI, V 2, Aug 20-25, 89, pp 972-978

Georgeff, Michael P. and Lansky, Amy L., "Procedural Knowledge", Proceedings of the IEEE, V 74, n 10, Oct 86, pp 1383-1391

Ginsberg, Matt, 1993, "Essentials of Artificial Intelligence", Morgan Kaufmann Publishers, Inc, San Francisco, CA

Hansen, R.J., Hall, D. L. and Kurtz, S.K., "A New Approach to the Challenge of Machinery Prognostics", Transactions of the ASME, V 117, Apr 95, pp 320-325

Harvey, J.J., "Expert Systems: An Introduction", International Journal of Computer Technology, V 1, n 1-2, 1988, pp 53-60

Helfman, Richard, Dumer, John and Hanratty, Timothy, "TED-Turbine Engine Diagnostics" Army Research Laboratory Report, Sep 95

Huang, Samuel H. and Zhang, Hong-Chao, "Artificial Neural Networks in Manufacturing: Concepts, Applications and Perspectives, IEEE Transactions on Components, Packaging and Manufacturing Technology - Part A, V 17, n 2, Jun 94, pp 212-224

Illi, Orlando J. Jr, Greitzer, Frank L., Kangas, Lars J. and Reeve, Tracy, "An Artificial Neural Network System for Diagnosing Gas Turbine Engine Fuel Faults" Presented at the 48th Meeting of the Mechanical Failures Prevention Group, Apr 19-21, 94

Jain, Anil K., Mao, Jainchang and Mohiuddin, K.M., "Artificial Neural Networks: A Tutorial", Computer, V 29, n 3, Mar 96, pp 31-44

Kangas, Lars J., Greitzer, Frank L., Illi, Orlando J. Jr, "TEDANN: Turbine Engine Diagnostic Artificial Neural Network" Presented at the American Defense Preparedness Association National Symposium, Mar 28-30, 94

Khan, Emdad and Rahman, Sayeed, "Fuzzy Logic Gets a Boost from Neural Nets", Electronic Design, V 43, n 13, Jun 26, 95, pp 128-132

Klir, George J., "Fuzzy Logic", IEEE Potentials, V 14, n 4, Oct/Nov 95, pp 10-15

Kuo, R. J., "Intelligent Diagnosis for Turbine Blade Faults Using Artificial Neural Networks and Fuzzy Logic", Engineering Applications of Artificial Intelligence, V 8, n 1, 1995, pp 25-34

Liebowitz, Jay, "Expert Systems: a Short Introduction", Engineering Fracture Mechanics, V 50, n 5/6, 1995, pp 601-607

McDonnell Douglas Information Sheet, "Maintenance and Repair Support System, MARSS"

Meher-Homji, Cyrus B., "A Feasibility Study of the Application of Artificial Intelligence Techniques for Turbomachinery Diagnostics", Presented at the Gas Turbine Conference and Exhibit, Mar 18-21, 1985, pp 1-8

Merrington, Graeme L., "Fault Diagnosis in Gas Turbines Using a Model-Based Technique" Transactions of the ASME, V 116, n 2, Apr 94, pp 374-380

Milne, R., "Artificial Intelligence for Online Diagnosis", IEEE Proceedings, V 134, n 4, Jul 97, pp 238-243

Monostori, L. and Barschdorff, D., "Artificial Neural Networks in Intelligent Manufacturing", Robotics and Computer-Integrated Manufacturing, V 9, n 6, 1992, pp 421-437

Moore, Kevin L., "Artificial Neural Networks", IEEE Potentials, V 11, n 1, Feb 92, pp 23-28

Myers, Ware, "Introduction to Expert Systems", IEEE Expert, V1, n 1, 1986, pp 100-109

Pettigrew, James L., "Artificial Intelligence for Turbine Engine Diagnostics", Aerospace Engineering, V 15, n 2, Mar 95, pp 9-13

Schofield, Julie A., "Fuzzy Logic Offers New Ways to Solve Tough Control Problems", Design News, V 50, n 13, Jul 10, 95, pp 23-24

Shandle, Jack. "Fuzzy Logic is Shedding Its 'Half-Baked' Image", Electronic Design, V 42, n 6, Mar 21, 94, pp 75-80

Somers, Richard D., "Engine Monitoring System Diagnostics for the TF30-P111 Engine", Autotestcon Proceedings, 1990, pp 567-573

Sondak, Norman E. and Sondak, Vernon K., "Neural Networks and Artificial Intelligence", SIGCSE Bulletin, V 21, n 1, Feb 89, pp 241-244

Taylor, Malcom S. and Monyak, John T., "Statistical Analysis of Turbine Engine Diagnostic (TED) Field Test Data", Army Research Laboratory Report, Nov 94

Textron Lycoming AGT 1500 Gas Turbine Informational Brochure, 1990

Vankayala, Vidya Sagar S. and Rao, Nutakke D., "Artificial Neural Networks and Their Applications to Power Systems - A Bibliographical Survey", Electric Power System Research, V 29, n 1, Oct 93, pp 67-69

18 Nov 96

To Whom it May Concern,

My name is Captain Mark J. Pincoski, 054-42-8509, of the U. S. Army. I have recently completed my Master's work at the Pennsylvania State University in Mechanical Engineering. I am enclosing a copy of my completed thesis.

MARK J. PINCOSKI
CPT, AD
U. S. Army